

# We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index  
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?  
Contact [book.department@intechopen.com](mailto:book.department@intechopen.com)

Numbers displayed above are based on latest data collected.  
For more information visit [www.intechopen.com](http://www.intechopen.com)



---

# Highly Deforming Computational Meshes for CFD Analysis of Twin-Screw Positive Displacement Machines

---

Sham Rane, Ahmed Kovačević, Nikola Stošić and Ian Smith

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.71885>

---

## Abstract

Commercial flow solvers can be used to obtain flow solutions in applications with deforming domains, but, in general, are not suitable for screw machine flow calculations. This is due to the large magnitude of deformation of the domain and the geometrical complexity of helical rotors. In this chapter, the governing equations for deforming domains and three methods of obtaining mesh movement, commonly used by FVM solvers, have been analysed. A comparative study of customised methods of grid generation for screw machines, using algebraic and differential approaches, is shown to help in the selection of techniques that can improve grid quality, robustness and speed of grid generation. The analysis of an oil-injected twin-screw compressor is included as a test case to demonstrate the application of SCORG, a deforming grid generator, as a means of predicting performance.

**Keywords:** finite volume method, deforming grid generation, arbitrary Lagrangian-Eulerian mesh motion, positive displacement machines, twin-screw compressors, twin-screw expanders

---

## 1. Introduction

Rotary screw compressors and expanders are positive displacement machines with two meshing rotors in the form of helical lobed screws. The working principle of these machines is simple. The male rotor engages with the female rotor forming a closed space with the housing, which forms the working chamber within a casing and its formation, and volumetric change, with rotation, is shown in **Figure 1**.

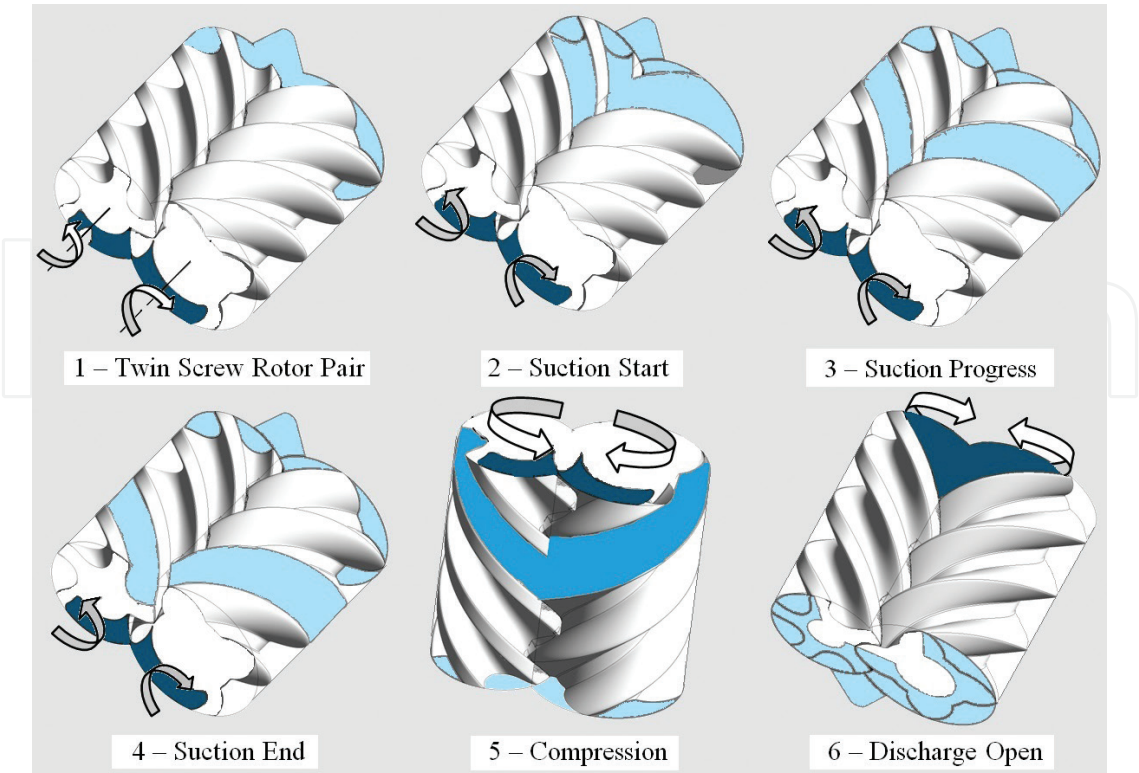


Figure 1. Operating stages in a twin-screw compressor.

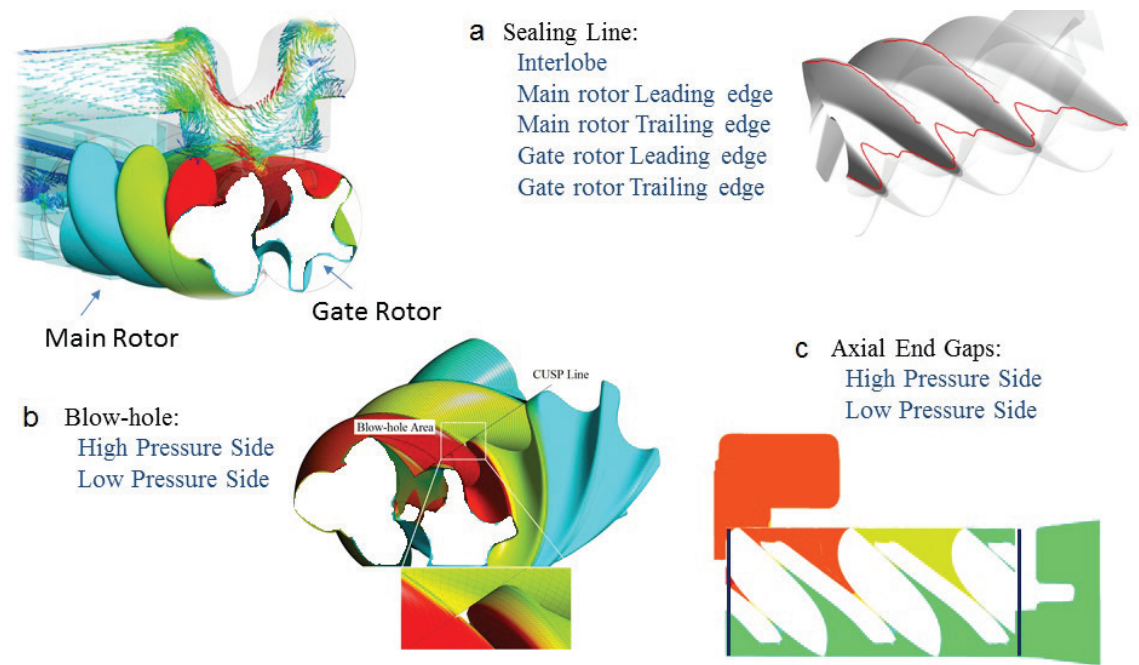


Figure 2. Leakages in a twin-screw compressor to be captured by computation mesh.

The compression process takes place between positions 4 and 6, and its analysis, by means of computational fluid dynamics (CFD), is challenging due to the existence of very small fluid leakage paths within the working chamber, associated with the volume change, the coexistence of both the working fluid and its lubricant, or coolant, as well as fluid injection and, most

importantly, the lack of methodologies available to generate meshes required for full three-dimensional transient simulations.

**Figure 2(a–c)** shows the various leakage gap flows that are required to be accurately predicted by the flow solver [6]. An essential requirement for the computational mesh is to capture these gaps accurately. In the core region of the compression chamber, the size is of the order of  $1000\times$  as compared to the gaps, and all these regions are dynamically deforming.

Hence, only numerical solvers, operating with customised grid generators, have been used successfully to analyse the processes within these machines. Accordingly, in the next section, the governing equations for deforming domains and methods to analyse mesh movement that handle large mesh deformations with high-solution accuracy are reviewed, and a **user-defined nodal displacement** procedure, using external grids, derived from them is given in detail.

## 2. CFD calculations using arbitrary Lagrangian-Eulerian formulation with FVM for deforming control volumes

For a general variable  $\phi$ , which can be a scalar or vector quantity, the integral form of the conservation equation, as given by Ferziger and Perić [10], can be represented by the general transport in Eq. (1):

$$\underbrace{\frac{\partial}{\partial t} \int_{\Omega} \rho \phi d\Omega}_{\text{transient}} + \underbrace{\int_S \rho \phi \mathbf{v} \cdot \mathbf{n} dS}_{\text{convection}} = \underbrace{\int_S \Gamma \text{grad } \phi \cdot \mathbf{n} dS}_{\text{diffusion}} + \underbrace{\int_{\Omega} q_{\phi} d\Omega}_{\text{source}} \quad (1)$$

When the control volume is deforming, or not fixed in space, the  $\Omega$  domain changes with time, due to movement of the boundaries. This movement is defined either as a function of time or as dependent on the current solution field. The convective fluxes such as the mass flux are calculated in such cases, using relative velocity components at the cell faces. If the coordinate system remains fixed and Cartesian velocity components are used, the only change in the conservation equations is the appearance of the relative velocity  $(\mathbf{v} - \mathbf{v}_b)$  in all convective terms, where  $\mathbf{v}_b$  is the velocity vector at the cell's face.

The application of Leibnitz's rule, given in Eq. (2), to differentiate the integral transient term of Eq. (1), results in Eq. (3) that gives the integral form of a general conservation equation in **arbitrary Lagrangian-Eulerian (ALE)** formulation:

$$\frac{\partial}{\partial t} \int_{\Omega} \rho \phi d\Omega = \frac{d}{dt} \int_{\Omega} \rho \phi d\Omega - \int_S \rho \phi \mathbf{v}_b \cdot \mathbf{n} dS \quad (2)$$

$$\underbrace{\frac{d}{dt} \int_{\Omega} \rho \phi d\Omega}_{\text{transient}} + \underbrace{\int_S \rho \phi (\mathbf{v} - \mathbf{v}_b) \cdot \mathbf{n} dS}_{\text{convection}} = \underbrace{\int_S \Gamma \text{grad } \phi \cdot \mathbf{n} dS}_{\text{diffusion}} + \underbrace{\int_{\Omega} q_{\phi} d\Omega}_{\text{source}} \quad (3)$$

This is arbitrary, because the grid velocity  $\mathbf{v}_b$  and grid motion are independent on the fluid motion. However, when the cell faces move, the conservation of mass and other conserved quantities are not necessarily ensured if the grid velocities are calculated explicitly and, in turn, are used to calculate the convective fluxes. To ensure that these equations are completely conserved, the space conservation law needs to be satisfied. This is given by

$$\frac{d}{dt} \int_{\Omega} d\Omega + \int_S \mathbf{v}_b \cdot \mathbf{n} dS = 0 \quad (4)$$

Space conservation can be regarded as mass conservation with zero fluid velocity. The unsteady terms involving integrating the control volume  $\Omega$ , in the governing equation, which is now changing with time, need to be treated to satisfy the space conservation equation with a deforming and/or moving grid. If the implicit first-order Euler scheme is used for temporal discretisation, the transient term can be discretised as

$$\frac{d}{dt} \int_{\Omega} \rho \phi d\Omega = \frac{(\rho \phi \Omega)^{n+1} - (\rho \phi \Omega)^n}{\Delta t} \quad (5)$$

where  $n$  and  $n+1$  represent the current and next time levels, respectively. The  $n+1$  volume  $\Omega^{n+1}$  is computed from

$$\Omega^{n+1} = \Omega^n + \frac{d\Omega}{dt} \Delta t \quad (6)$$

where  $d\Omega/dt$  is the rate of change of volume of the CV. In order to satisfy Eq. (4), this is calculated as

$$\frac{d\Omega}{dt} = \int_S \mathbf{v}_b \cdot \mathbf{n} dS = \sum_j^{n_f} \mathbf{v}_{b,j} \cdot \mathbf{S}_j \quad (7)$$

where  $n_f$  is the number of faces on the control volume and  $\mathbf{S}_j$  is the  $j$ th face area vector.

The dot product  $\mathbf{v}_{b,j} \cdot \mathbf{S}_j$  on each control volume face is calculated as  $\partial\Omega_j/\Delta t$ , from the volume swept out  $-\partial\Omega_j$  by that face over the time step  $\Delta t$ . Therefore, the mass flux  $\dot{m}_j$  can be calculated using  $\partial\Omega_j/\Delta t$  instead of being explicitly calculated from the grid velocity  $\mathbf{v}_{b,j}$ :

$$\dot{m}_j = \int_{S_j} \rho (\mathbf{v} - \mathbf{v}_{b,j}) \cdot \mathbf{n} dS_j \approx \rho_j (\mathbf{v} \cdot \mathbf{n})_j S_j - \frac{\rho_j \partial\Omega_j}{\Delta t} \quad (8)$$

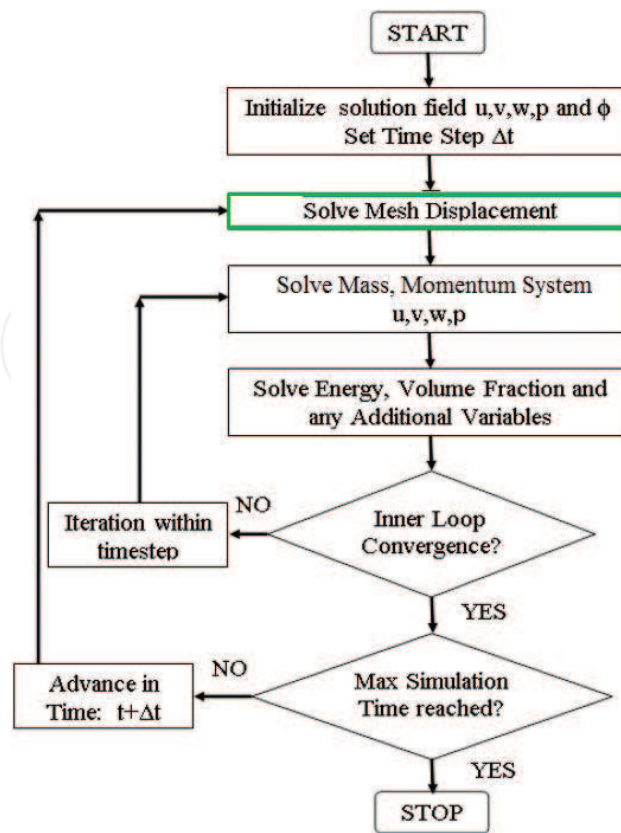
The requirement for space conservation in flow equations at moving integration points was introduced by Trulio and Trigger [11] when applied to a finite difference calculation framework. If the volume change and mass fluxes are calculated as given in Eq. (8), then this is ensured. Thomas and Lombard [12] presented the implementation of space conservation for a density-based finite difference scheme on structured meshes. Gosman [13] presented applications of this to reciprocating in-cylinder problems with moving boundaries.

When FVM is applied to solve for flows in screw machines, it faces a major challenge in the grids required for transient simulations. These grids have to capture the complex geometry formed by the fluid enveloping the screw rotors, and at the same time, they need to change their shape by large orders of magnitude. These domains are treated as highly deforming grids.

## 2.1. Solution of governing equations

**Figure 3** represents a flow chart of the solution process for a FVM with deforming domains. The solver used in this analysis is a pressure-based coupled solver, ANSYS CFX. For compressible flows, where the density varies, an equation of state is used to obtain the relationship





**Figure 3.** Flow chart of solution process with deforming domains.

between density and pressure, and the mass conservation equation is linearised in terms of pressure. Hence, a coupled system of equations, involving three Cartesian velocity components and pressure, is solved for every iteration in a time step.

The integration process and the solution of the governing equations ensure that the space conservation is retained at the new time instance. The highlighted block, named ‘Solve Mesh Displacement’ in **Figure 3**, should allow for the movement of the initial grid, in time, to the new position, during which the control volume deforms. The mesh should retain the same number of nodes, cells and topology, while the cell deformation should be defined by the movement of its nodes. The node displacement in ALE approach can be solved using different strategies, for example, (a) spring smoothing, (b) diffusion equation smoothing, (c) user-defined nodal displacements through external subroutines or (d) key-frame re-meshing.

### 2.1.1. Diffusion equation-based mesh smoothing

In the diffusion equation-based mesh smoothing, the specified displacement on the boundary nodes is propagated to the interior nodes by solving Eq. (9) internally in the solver:

$$\nabla \cdot (\Gamma_{disp} \nabla \delta) = 0 \quad (9)$$

Here,  $\Gamma_{disp}$  is the mesh stiffness that depends on the local cell volumes and the distance of the nodes from their deforming boundaries. This method is presented in **Figure 4**.

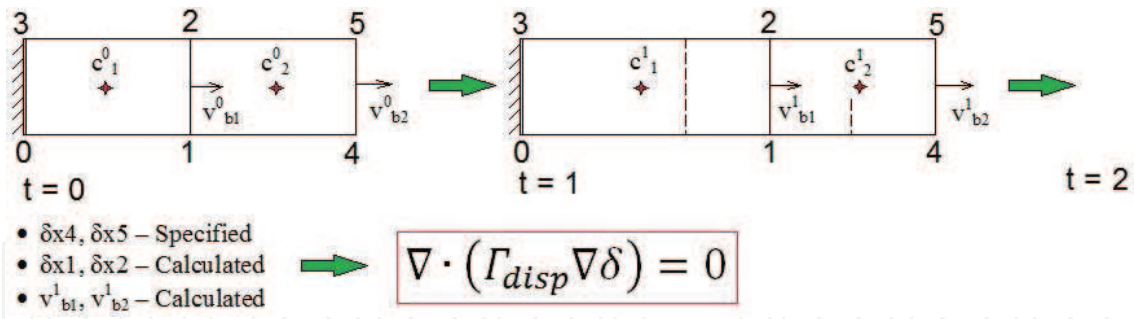


Figure 4. Grid deformation using diffusion equation mesh smoothing.

The number of cells, and their connectivity, remains constant for the duration of the calculation, with the grid being deformed at the beginning of each time step. A mesh smoothing method is not suitable for large boundary displacements because it produces cells with high skewness and there are chances of element failure due to negative volumes being formed after high nodal displacements. This limitation is dependent on the types of cell; the grid size, relative to the magnitude of the boundary displacement; the relative orientation of the cells; the boundary displacement; the position of the nondeforming boundaries, etc. For example, for tetrahedral types of cell, the skewness quality degrades faster with the boundary displacements, compared to hexahedral cells. Hence, for cases with simple geometries like a piston-cylinder combination, this method can be used, even with large displacements with hexahedral grids, but it fails for complex geometries, as in the case of screw compressors.

### 2.1.2. User-defined nodal displacement

Customised grid generators can be used to generate a set of grids representing nodal locations for every time step externally, prior to the numerical solution of flow in the calculation domain. The numerical grid is replaced at the beginning of each time step by the use of appropriate user subroutines, often called ‘junction boxes’. This method is called user-defined nodal displacement and is shown graphically in Figure 5.

In the principle, it achieves the same result as the diffusion equation mesh smoothing, but it is suitable for large boundary displacements because the nodal positions are controlled externally.

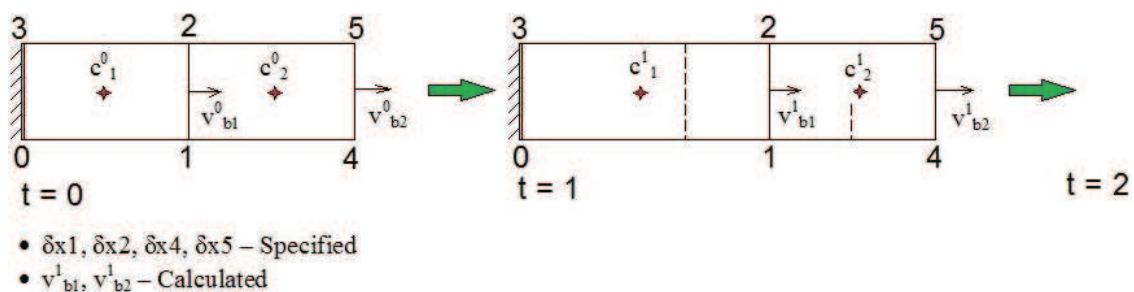
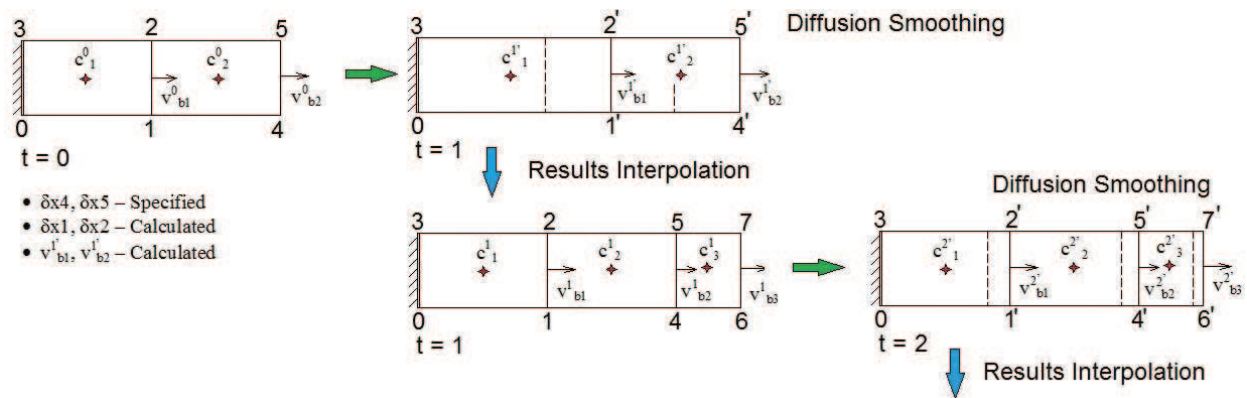


Figure 5. Grid deformation using user-defined nodal displacement.



**Figure 6.** Grid deformation using key-frame grid re-meshing.

### 2.1.3. Key-frame grid re-meshing

**Figure 6** shows another strategy suitable for high grid deformations which uses general-purpose grid generators.

After every time step, diffusion smoothing is performed to obtain the new geometry. A check is then performed to determine whether the control volume cells can absorb further deformation without generating negative volume elements. Upon the discovery of irregular cells, groups of cells are selectively re-meshed locally, retaining the boundary geometry but with a change in the number of cells and their connectivity. This process is called key-frame re-meshing. Re-meshing is followed by the next time step calculations, similar to those described in Section 2.1.1, but an intermediate interpolation of the solution is required after the previous time step, because, generally, all the cells will have changed connectivity.

In Section 2.2, a comparison of the solutions, derived from the methods of grid deformation described, has been presented, to study the accuracy of calculations, when applied to positive displacement flow applications.

## 2.2. Investigation of mesh deformation techniques

A case study is presented here to assess the applicability of the key-frame re-meshing method to the calculation of flow in positive displacement machines. A reversible adiabatic compression process in a piston cylinder was solved by the use of three grid deformation techniques, namely diffusion smoothing, user-defined nodal displacements and key-frame grid re-meshing. Such a simplistic geometry and thermodynamic process was chosen because it forms the fundamental mechanism for calculating performance by CFD in positive displacement machines and analytically derived results are available for error estimation. The isentropic compression-expansion process in a reciprocating piston cylinder can be modelled by a polytropic process equation relating the gas pressure and volume. Since the process is both adiabatic and reversible, there will be no energy losses or gains in the control volume, and the gas will return to its initial state given by Eq. (10):

$$\left(\frac{p_2}{p_1}\right) = \left(\frac{V_1}{V_2}\right)^\gamma = \left(\frac{T_2}{T_1}\right)^{\frac{\gamma}{\gamma-1}} \quad (10)$$

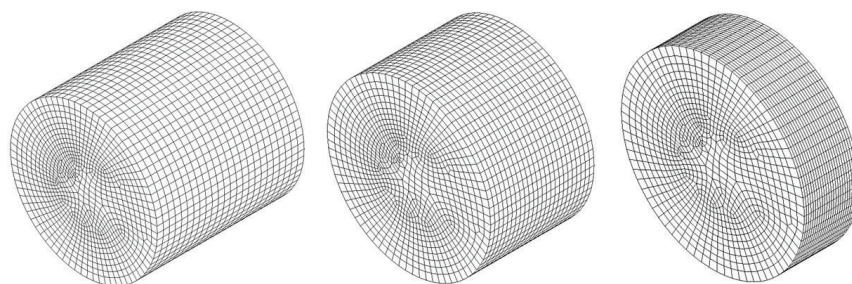


Consider a hypothetical piston cylinder arrangement with a sinusoidal displacement given to the piston. For this trial, a cylinder with diameter 100 mm and length 100 mm was considered. The initial position of the piston was for a maximum volume of  $7.854 \times 10^{-4} \text{ mm}^3$ . The piston displacement was 70 mm, varying sinusoidally with a frequency of 50 Hz. The final minimum cylinder volume was  $2.356 \times 10^{-4} \text{ mm}^3$ . This gave a fixed volume ratio of 3.333 for the system. Based on Eq. (10), for an initial absolute pressure in the cylinder of 2.013 bar, the expected peak pressure is 10.86 bar. Similarly, for an initial temperature of 298 K, the expected peak temperature is 482.35 K. In order to make the CFD model isentropic, all boundaries were defined as adiabatic, and the calculation of the viscous dissipation term, in the conservation equation of total energy, was turned off. This took reversibility, due to the turbulent viscosity dissipation factor, into account. Molecular viscosity and no-slip conditions at the walls were retained, for the formulation to be close to physical conditions.

The most suitable configuration for diffusion smoothing is a hexahedral mesh since the cell quality does not deteriorate much when boundaries deform. **Figure 7** shows the hexahedral mesh generated by diffusion smoothing at three different time steps. The hexahedral meshes generated by diffusion smoothing were identified as SH. The mesh shown in **Figure 7** is coarse, consisting of 22,752 cells identified as SH1. Two refined meshes were generated by the same method and named SH2 consisting of 49,962 cells and SH3 consisting of 74,720 cells (**Table 1**). For the key-frame-based re-meshing, a tetrahedral mesh was selected, as shown in **Figure 8**. Three different cases were generated, namely coarse KR1, medium KR2 and fine KR3 consisting of an equal number of cells and nodes, as shown in **Table 1**. For the user-defined nodal displacement, a hexahedral mesh was selected, and three grid sizes were generated to correspond to sizes of the diffusion smoothing. These were named UH1, UH2 and UH3, respectively, as listed in **Table 1**.

The time step was  $2.8571 \times 10^{-4} \text{ s}$ , while each cycle contained 70 time steps which corresponded to 50 cycles per second. Mesh deformation was applied after each time step. The implicit second-order backward Euler discretisation was used within the pressure-based coupled solver. The advection scheme was high resolution, and the turbulence model was  $\kappa$ - $\epsilon$ . An rms residual target of  $1.0 \times 10^{-4}$  was maintained for all the equations. The fluid was air, assumed to be an ideal gas, with a molar mass of  $28.96 \text{ kg kmol}^{-1}$ , specific heat capacity of  $1004.4 \text{ J kg}^{-1} \text{ K}^{-1}$ , dynamic viscosity of  $1.831 \times 10^{-5} \text{ kg m}^{-1} \text{ s}^{-1}$  and thermal conductivity of  $2.61 \times 10^{-2} \text{ W m}^{-1} \text{ K}^{-1}$ .

**Figure 9** shows the variation of the piston displacement with time for the same volume ratio in cases SH1 and KR1. **Figure 10** shows the change of pressure with time in cases SH1 and KR1.



**Figure 7.** Hexahedral mesh using diffusion smoothing (SH1).

	Cycle	SH1	SH2	SH3	KR1	KR2	KR3	UH1	UH2	UH3
Number of cells		22,752	49,962	74,720	42,098	118,749	211,418	22,752	49,962	74,720
Number of nodes		24,650	53,244	79,171	11,849	31,514	53,148	24,650	53,244	79,171
Error in pressure [%]	I	-0.44	-0.44	-0.44	1.17	1.34	1.11	-0.44	-0.44	-0.44
	II	-0.42	-0.42	-0.42	6.23	7.26	7.41	-0.42	-0.42	-0.42
	III	-0.41	-0.41	-0.41	10.55	12.95	14.41	-0.41	-0.41	-0.41
Error in temperature [%]	I	-0.08	-0.08	-0.08	0.05	0.32	0.25	-0.08	-0.08	-0.08
	II	-0.07	-0.07	-0.07	3.44	4.18	3.74	-0.07	-0.07	-0.07
	III	-0.05	-0.05	-0.05	8.56	9.80	10.24	-0.05	-0.05	-0.05
Error in mass [%]	I	-0.38	-0.32	-0.30	1.78	1.91	2.27	-0.38	-0.32	-0.30
	II	-0.38	-0.32	-0.30	2.91	3.21	3.88	-0.38	-0.32	-0.30
	III	-0.38	-0.32	-0.30	2.15	3.29	4.43	-0.38	-0.32	-0.30

Table 1. Error in pressure, temperature and gas mass.

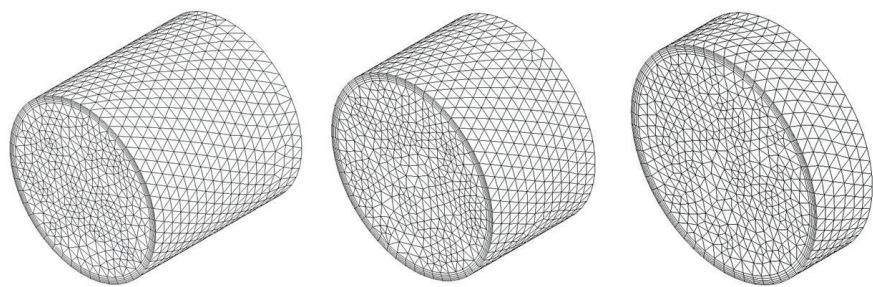


Figure 8. Tetrahedral mesh used in key-frame re-meshing (KR1).

The results obtained with user-defined nodal displacement UH1 were identical to that of SH and hence are not plotted. All cases, grid details and results are presented in **Table 1**. Although both diffusion smoothing (SH1) and key-frame re-meshing (KR1) follow the same change of volume during compression and expansion, the calculated peak pressures are not equal. In the case of diffusion smoothing, the pressure in the first cycle achieved the theoretical peak of 10.86 bar and consistently repeated itself in the following cycles. But in the case of key-frame re-meshing, the maximum pressure in the first cycle was higher than the theoretical value, and it continued to increase in the following cycles. Similarly, the initial state of pressure at the end of expansion did not return to its base value as it did in the case of diffusion smoothing.

**Figure 11** shows the temperature change with time in the cylinder for both cases SH1 and KR1. The peak temperatures are not equal in the two. In the case of diffusion smoothing, the temperature in the first cycle rises to the theoretical peak of 482.35 K and repeats itself in the following cycles consistently. In the case of key-frame re-meshing, the peak temperature in the first cycle is similar to the theoretical value, but it continues to increase in the following

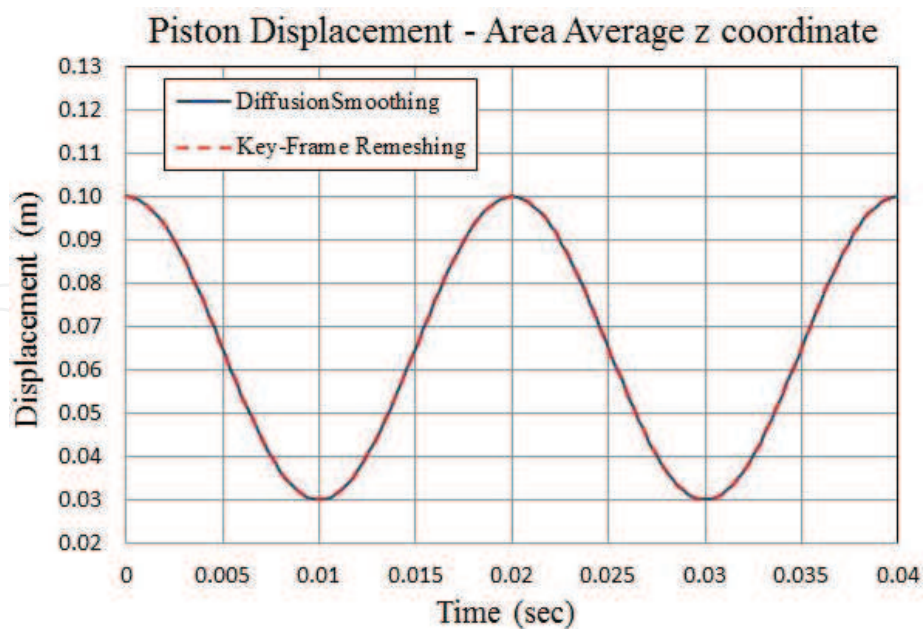


Figure 9. Piston displacement plot.

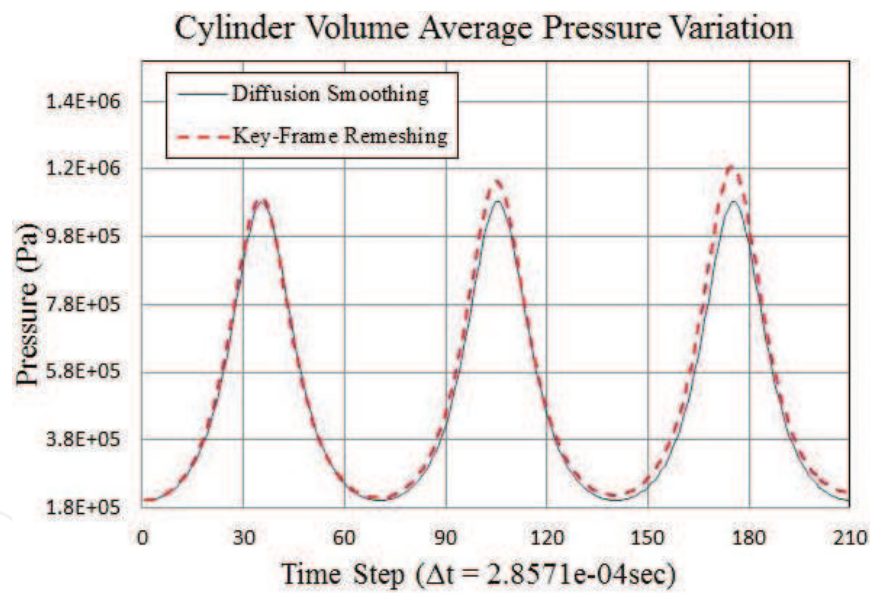


Figure 10. Cyclic temperature history.

cycles. Similarly, the initial state of temperature, at the end of expansion, does not return to its base level of 298 K as it does in the case of diffusion smoothing. **Table 1** shows the error in the prediction of pressure and temperature obtained from the three different grid deformation strategies with three different grid sizes and over multiple consecutive compression cycles.

The error in the pressure predictions over three consecutive compression cycles for the various grid deformation strategies is shown in **Figure 12**. The results show that the diffusion smoothing (SH)-based method of grid displacement and the user-defined nodal displacement (UH)-based method of grid displacement produce the same, highly accurate predictions which

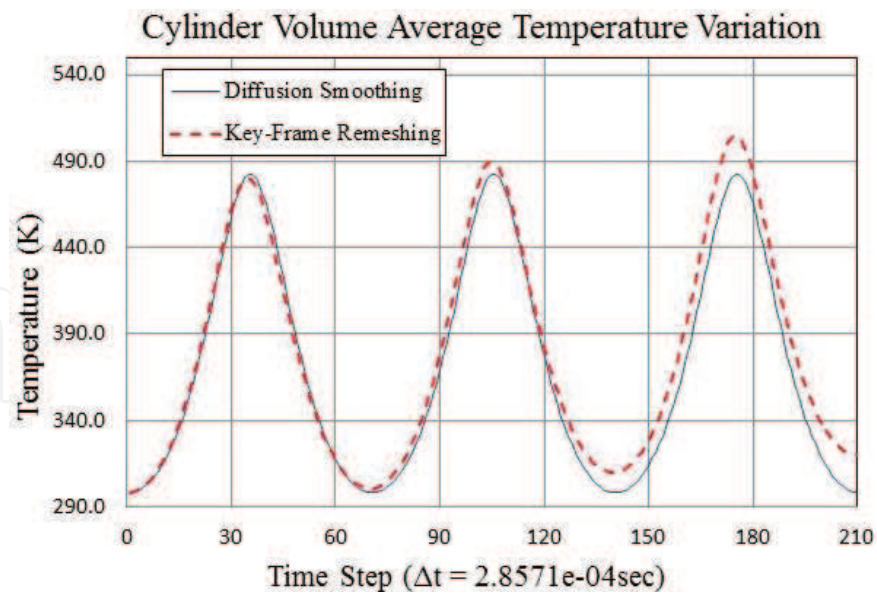


Figure 11. Cyclic pressure history.

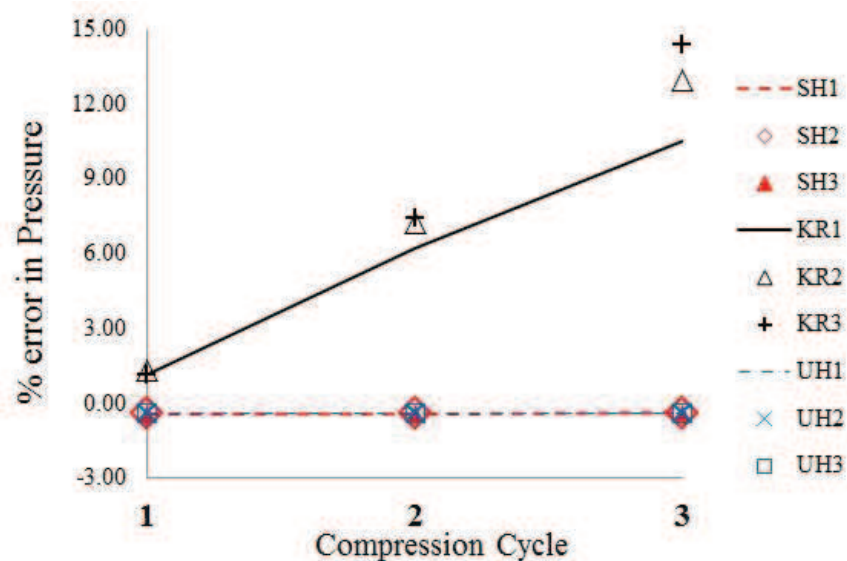


Figure 12. Error in pressure predictions of consecutive adiabatic cycles.

conform with the theoretical results for this highly deforming boundary formulation. However, errors in the pressure and temperature prediction exist when the key-frame re-meshing-based method of grid deformation is used. For the key-frame re-meshing method, the error increases with the grid refinement. Since the mesh is replaced for each time step, it is quite possible that it induces an increase in calculation error. As pointed out by Ferziger and Perić [10], this can lead to artificial mass source errors in the continuity equation that can also accumulate with flow time. **Table 1** shows the error in mass over multiple consecutive compression cycles for the three different grid deformation strategies. Limiting the mesh, to be replaced, only when the cell quality is severely reduced, should help to reduce the error, but in the case of complex topologies, as in screw machines, this is very difficult. In conclusion, the key-frame re-meshing



method gave inconsistent results with increasing error in pressure and temperature predictions in successive compression cycles. Grid refinement also did not improve the accuracy of key-frame re-meshing, while reducing the time step made excessive demands on preprocessing. However, diffusion smoothing and user-defined nodal displacement produced identical results with accurate estimates of pressure and temperature and hence are the preferred approaches to deal with highly deforming domains such as twin-screw machines.

### 2.3. Algebraic grid generation of highly deforming screw rotor grids for user-defined nodal displacement

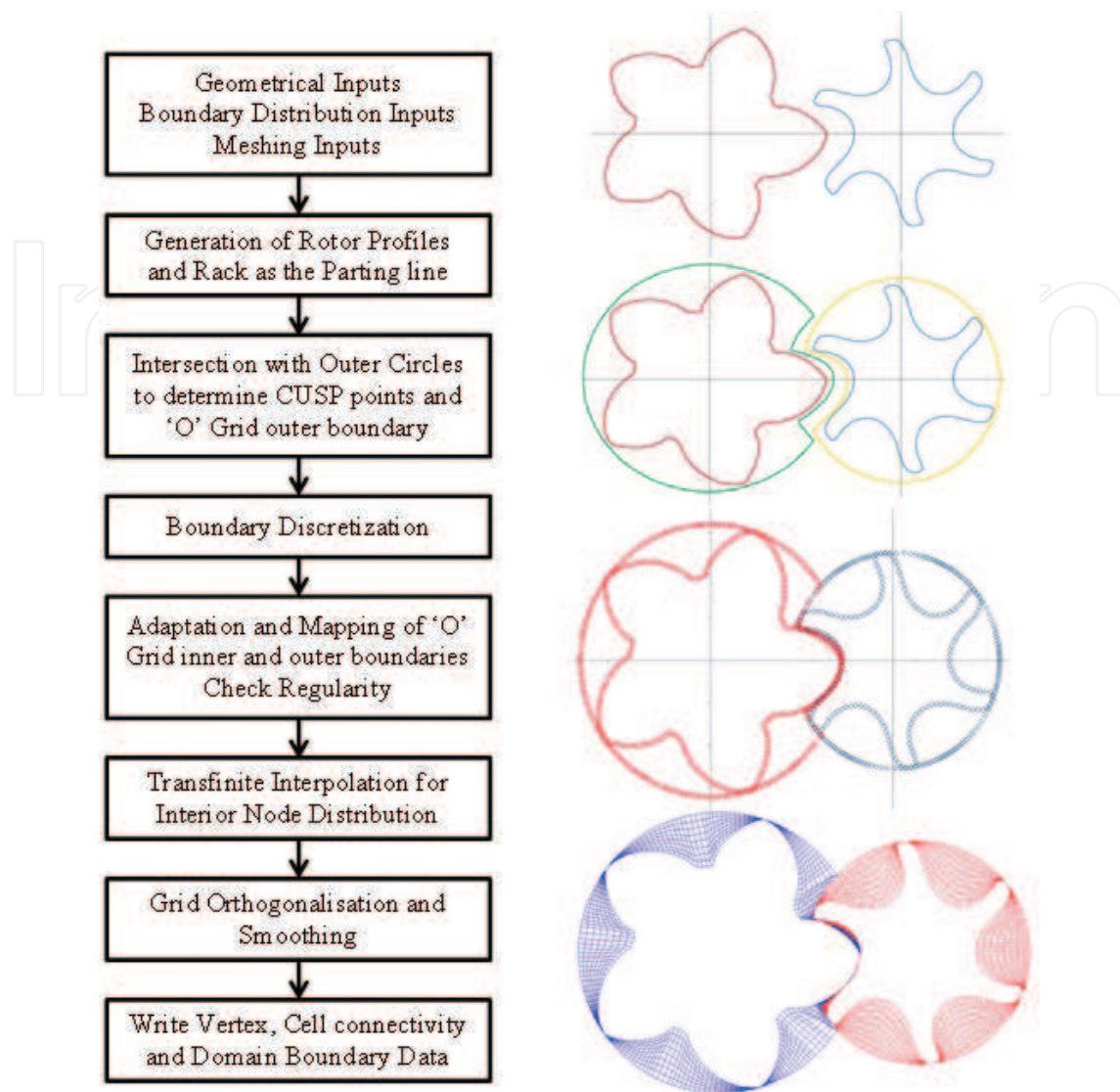
**Figure 13** shows the main stages of the procedure developed by Kovačević et al. [5, 6]. The domain of the working chamber is first decomposed into the low-pressure port, the rotor domain and the high-pressure port. The rotor domain, which deforms with time, consists of a hexahedral grid, and a set of grids representing the entire compression cycle are required to be supplied to the solver for continuous simulation. The low-pressure and high-pressure ports can be meshed using general-purpose grid generators, and it is easier to have a tetrahedral grid with fine prism layers covering the boundaries. These tetrahedral grids interface with a structured grid, built on the rotor domain via nonconformal conservative interfaces. An algebraic grid generation method employing multiparameter one-dimensional adaptation and transfinite interpolation is used to construct the hexahedral grid in the rotor domain. A block-structured numerical mesh is desired in the rotor domain. In order to construct an O grid, it is necessary to subdivide the topology of the male rotor, the female rotor and the casing together into two.

One O grid is constructed on the male side and the other on the female side. It is most convenient to do this splitting in 2D cross section of the rotors, and the envelope method of gearing is used to construct a rack curve between the male and the female rotor profiles to split into two O grid primitives as shown in **Figure 13**. Consecutive 2D cross sections are calculated individually, as shown by Kovačević [5]. This procedure is the most important function of the framework and provides flexibility to extend it to more complex rotor designs like variable pitch or variable profiles. A detailed description of the algebraic approach for variable screw rotors is given by Rane [4].

### 2.4. Differential approach for generation of highly deforming screw rotor grids for user-defined nodal displacement

Differential equation-based decomposition is a generic method that can be applied to a variety of twin rotor machines like tooth compressors, screw compressors, etc. The working chamber is again split into the deforming domain of the rotors and the static domains of the low-pressure and high-pressure ports. The compression chamber is defined using the rotor profile and the casing, and a block structure grid is generated by the stages shown in **Figure 14** [3]. An FVM can be used to solve a Laplace equation system in the 2D domain, so formed. Since the rotor profiles have close tolerances, the region formed between the casing, and the two profiles have a very complex boundary. Hence, discretisation, using unstructured triangular cells, is the most appropriate. The gaps can be highly refined to capture, accurately, the iso-potential lines conforming with the boundary. **Figure 14** shows a triangular mesh generated using the commercial





**Figure 13.** Simplified block diagram: algebraic rotor grid generation.

grid generator ANSYS. In practice, 10–15 times higher refinement is necessary. Vande Voorde et al. [3] used a customised Delaunay triangulation program, as given by Riemsdagh et al. [14], for this initial triangular grid construction. Further details of this approach are given in [3].

The beneficial properties of the differential solution are well known and were first utilised by Crowley and Winslow [9] for generating a triangular 2D mesh in the numerical solution of PDEs for static magnetic fields. With the differential approach, the grid for the main flow calculations is, in turn, generated from the solution of an elliptic PDE like the Laplace equation  $\nabla^2 \phi = 0$ . Hence, the first task, after representing the compression chamber, is to slice it into a number of 2D cross sections. Depending on the lead of the helical rotor, each 2D section spacing is such that it corresponds to a unit degree of rotation of the main rotor. This allows for smooth representation of the rotor lead as well as physical rotation of the rotors during simulation. **Figure 14** shows the splitting curve or division line obtained in a 2D cross section of the compression chamber.

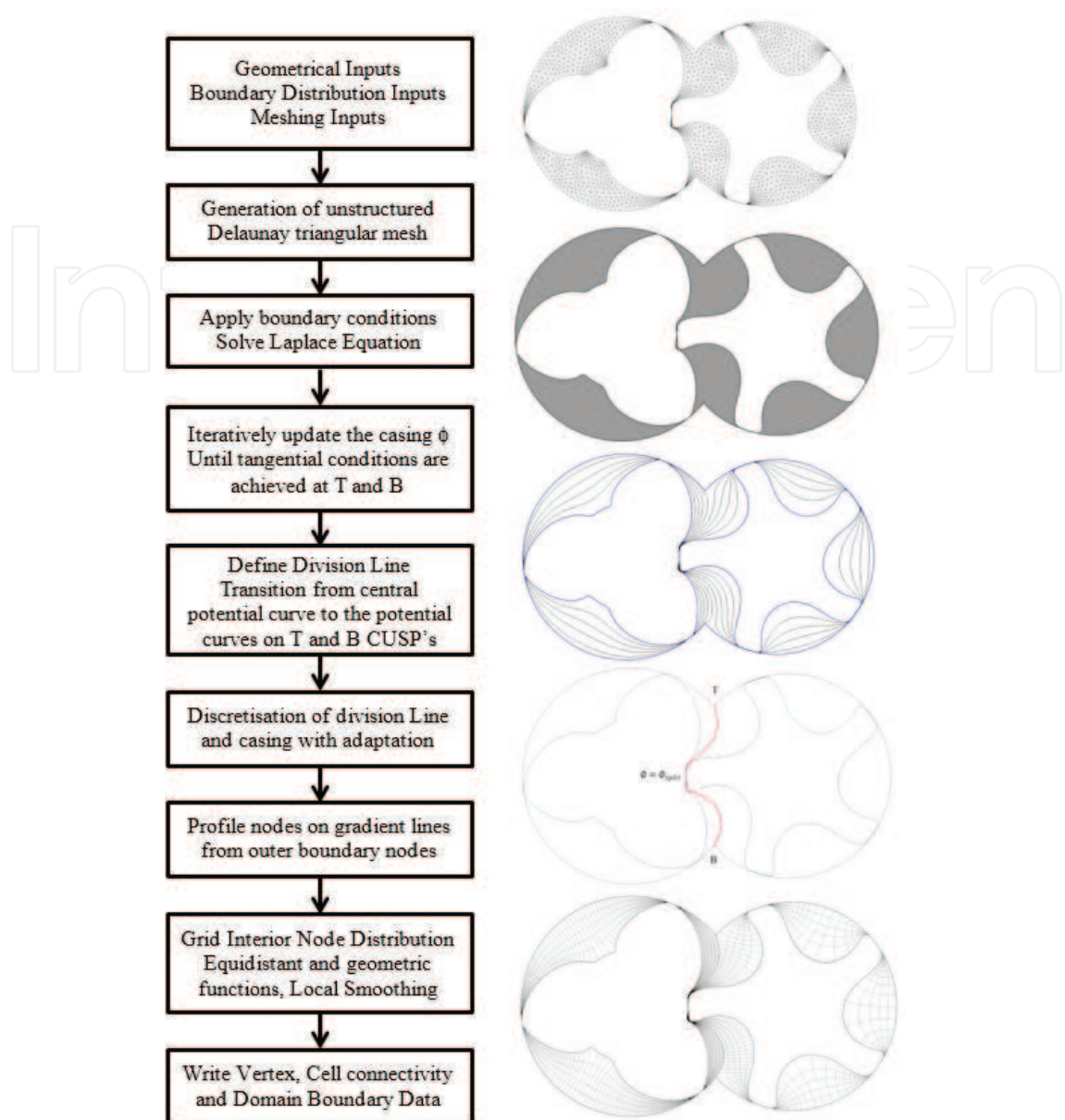


Figure 14. Simplified block diagram: differential rotor grid generation.

2.5. Comparison of the algebraic and differential approaches

Table 2 presents a comparison between the algebraic and the differential approaches applied to screw rotor grid generation. Each approach has its own advantages and limitations.

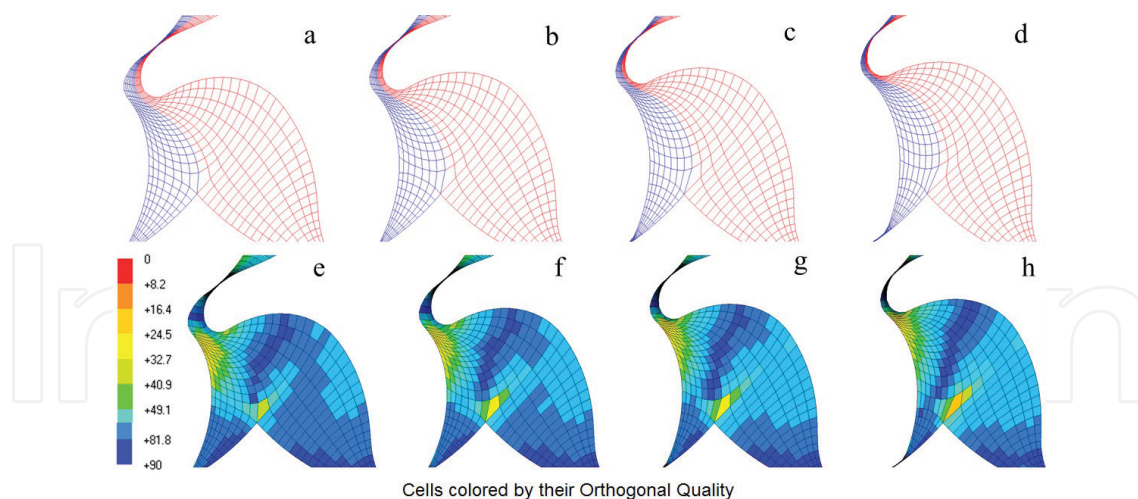
From the description of the techniques in the algebraic and differential approaches and the comparison of the two, it can be said that a good choice of theories is currently available for extending the grid generation to improve the quality, robustness and speed.

2.6. Combination of algebraic and differential grid generation approach

Algebraic methods are favourable for their speed of grid generation. In order to improve the resulting grid quality, the generated algebraic mesh can be used, and the partial differential

	Criteria	Algebraic approach [6]	Hybrid differential approach [3]
1	Rotor domain decomposition	Analytical rack or numerical rack Rack-based decomposition is very fast	Differential division line Differential division line decomposition is generic in nature
2	Boundary distribution	Profile discretisation according to selected adaptation Rotor projection used with regularity checks for outer boundary Rotor $\Rightarrow$ outer boundary With rotor $\Rightarrow$ outer boundary, the O blocks rotate with the rotor. Grid deformation occurs at the rack interface	Casing plus division line distributed first using equidistance and radius of curvature factors. Profile nodes projected of gradient line Outer boundary $\Rightarrow$ rotor With outer boundary $\Rightarrow$ rotor, the O blocks do not rotate but deform as rotor surface turns
3	Interface between the two O blocks	Nonconformal Nonconformal interface is handled by flux balance in the solver and gives more flexibility for adaptation	Conformal Conformal interface is desirable for higher-solution accuracy Demanded by some CFD solvers
4	Control on grid size	Circumferential and radial divisions and rotor angular positions Fixed number of angular positions constraint the solver time step size for the given speed	Circumferential and radial divisions and rotor angles at every degree position Intermittent rotor positions obtained by linear interpolation
5	Control functions	Two variable unidirectional adaptations on the rotor, TFI with tension spline factor, orthogonalisation and smoothing factors Rotor profile adaptation captures the 3D surface more accurately	Division line to CUSP transition, local smoothing and geometric spacing in radial direction
6	Time required for grid generation	For 50 rotor positions, on an i3 processor, it takes about 5 s for boundary distribution and 30 s for 2D mesh generation Full set with 750 grids takes about 1 h	A representative mesh took about 900 s for one position on DS20 processor. Full set took about 15 h Solution to the initial PDE is iterative and consumes time to get tangential conditions at the CUSPs
7	Cell structure	In order to retain the CUSP line in 3D, some pyramid or prism cells are introduced by merging nodes at the CUSP	All cells are purely hexahedral. Near tip of the gate rotors some face Warpage occurs due to grid line alignments
8	Cell jump	Generally, near the CUSP region between the two O blocks	Generally, in the transition region from central division line to casing
9	Adaptation to variable lead rotors	Directly possible with 3D grid rearrangement [8]	Directly possible with 3D grid rearrangement
10	Adaptation to variable profile rotors	Possible with modification in the algorithm to execute 2D generation on each profile and with 3D grid rearrangement [8]	Practically, challenging due to the requirement of potential solution in 3D or on multiple 2D profiles for each position
11	Advantages	Fast and good control over the node distribution Rotor profile smoothly captured in the grid Grid refinement is relatively easy to be achieved	Very smooth cells Fully hexahedral Conformal interface between O blocks makes the grid suitable for a variety of solvers and complexities like multiphase flows
12	Difficulties	Due to nonconformal interface, some solvers do not function with the helical rotor grids	No functional difficulties except for the computationally expensive generation procedure

**Table 2.** Comparison of algebraic and differential approach for screw rotor grids.



**Figure 15.** Improved grid transition and cell quality achieved by a combination of algebraic and differential approach.

equation (PDE) system can be solved over this grid. Rane et al. [1, 2] describe the procedure to produce an algebraic grid that consists of a single domain enclosing both the main and gate rotors. In such a domain, a further PDE system of the Poisson form is solved. The improvement in mesh transition and cell orthogonality quality, obtained by implementing a combined algebraic and differential grid generation, is shown in **Figure 15**. This algorithm is currently available in the SCORG grid generation tool and a test case to demonstrate its application to oil-injected screw compressor calculation presented in the next section [7].

### 3. Analysis of an oil-injected twin-screw compressor

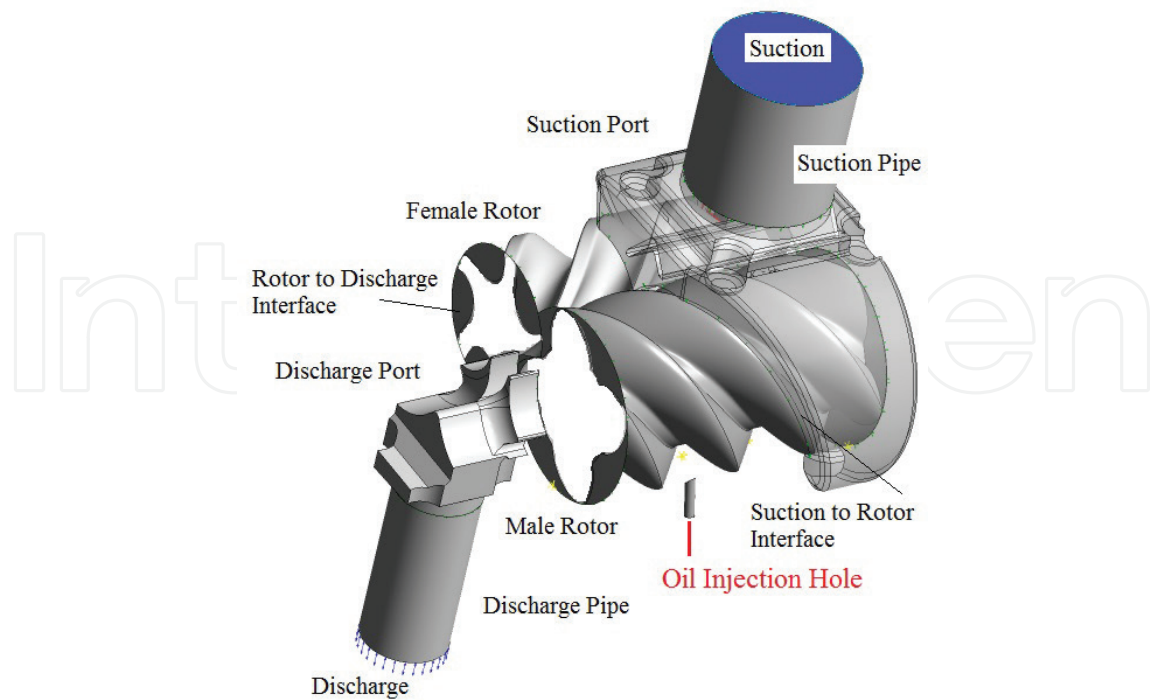
Flow in an oil-flooded twin-screw compressor with a combined axial and radial suction and an axial discharge port was calculated with a PDE grid, as shown in **Figure 15**. The CFD model setup in CFX solver is shown in **Figure 16**. The compressor with 'N' profile rotors has four male lobes and five female lobes. The nominal interlobe and radial and axial leakage gaps are  $60\text{ }\mu\text{m}$ . The operating speed of the machine is between 3000 and 6000 rpm, while discharge pressure can vary between 4.0 and 12.0 bar. A complete description of the model's numerical setup is given by Rane et al. [7]. Only discussion of the important results is presented here.

The quality of generated rotor mesh had a minimum orthogonal angle of  $6.5^\circ$ , a maximum expansion factor of 857 and a maximum aspect ratio 906 in less than 1% of the cells. The rotor grid node count was 467,992, while the total model's node count was 713,911. This includes the suction, discharge and oil injection ports.

#### 3.1. Pressure distribution

The variation of pressure in the compression chamber with abscissa as the main rotor rotation angle is shown in **Figure 17**. From the pressure angle plots, it can be observed that in all three cases the pressure during the compression part of the process aligns closely. At 3000 rpm when the discharge port opens, the flow pulsations in the port are low as compared to that at 6000 rpm. These increase with the increase in discharge pressure. Still for 3000 rpm, the

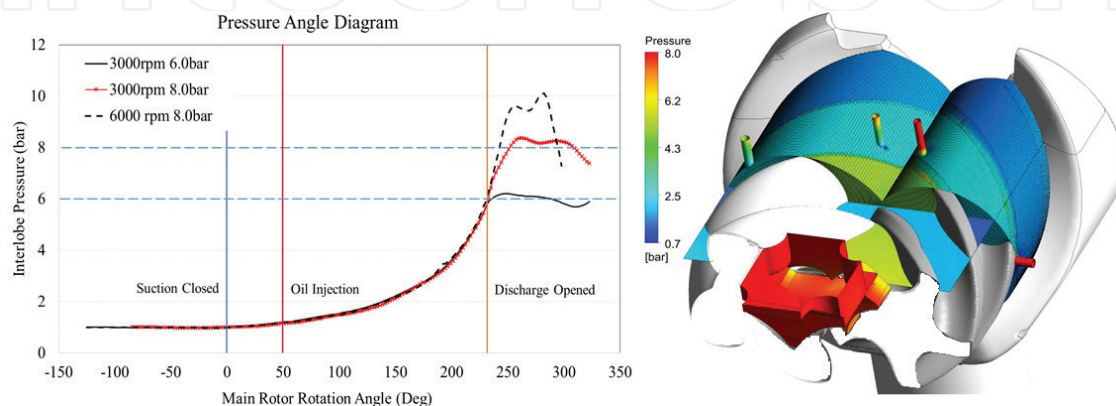




**Figure 16.** CFD model of an oil-injected twin-screw compressor in ANSYS CFX solver.

pulsations and the maximum pressure are both lower than at the higher speed. At 8.0 bar discharge pressure and 6000 rpm, the peak pressure reaches 10.0 bar. For the built-in volume ratio of 3.6, the 8.0 bar discharge pressure is an under-compression condition, also visible in the pressure curve. The peak over-compression at higher speed is due to throttling in the port.

A comparison of the performance predicted by CFD model with measured data is shown in **Figure 18**. Here, measurements at 6.0 bar and 3000 rpm have been used for normalisation. A close prediction of flow is seen at 6000 rpm. At 3000 rpm, the CFD model is under predicting flow at both 6.0 and 8.0 bar discharge pressures by about 10%. This could be due to the amount of injected oil in the CFD model, not being equal to that in the tests. By assuming a mechanical efficiency of 77%, the derived input power deviates from the measured values by less than 2% at both speeds and pressures.



**Figure 17.** Pressure diagram for three cases and distribution at 6000 rpm (8.0 bar discharge).



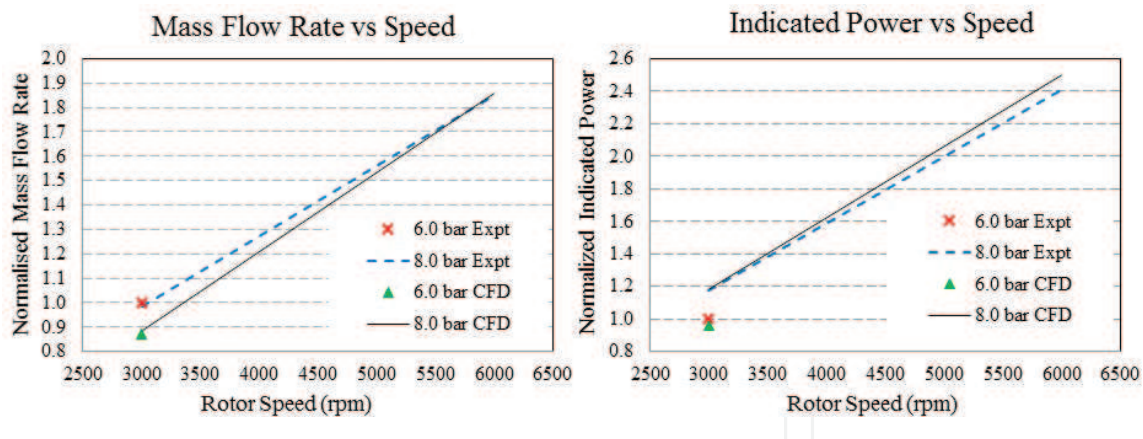


Figure 18. Performance comparison as normalised air flow and indicated power.

### 3.2. Oil distribution inside the compression chamber

The close interaction of the injected oil with the gate rotor lobes over one complete interlobe rotation is shown in **Figure 19(a–h)**. At position ‘a’, the formation of the oil spray begins when the injection hole is just exposed to the compression chamber. Its direction is influenced by the passing leading gate rotor lobe. The velocity of oil depends on the injection pressure. At ‘c’ oil film forms in the root of the rotor. At positions ‘d’, ‘e’ and ‘f’, the film spreads, the spray continues and, finally, intersection with the trailing lobe starts. At ‘g’ the hole is completely blocked by the trailing lobe width, and it follows, from the oil velocity vectors, that an oil flow pulse is formed. At ‘h’ the position of injection coincides with that at ‘a’, and the cycle repeats itself. A large vortex remains in the trailing working chamber.

The pulsating nature of the oil injection can also be seen in **Figure 20**. The quantity of oil injected at 8.0 bar is the same at both 3000 and 6000 rpm. In the analysis, the system pressure loss from separator vessel to the injection hole is assumed to be 0.5 bar in all the three cases.

For a more accurate model, an injection pressure measurement would be required that can be specified as a boundary condition. The oil flow rate at 6.0 bar is lower than that at 8.0 bar discharge pressure. The distribution of oil on gate rotor surface and its interaction in the discharge

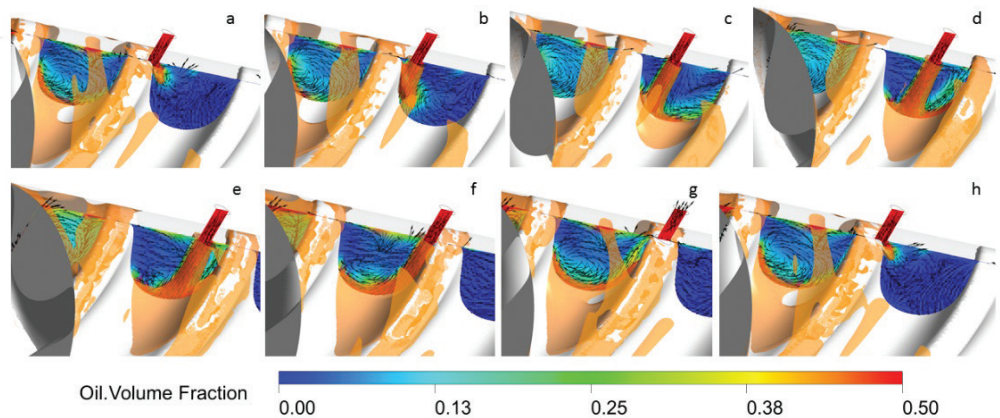
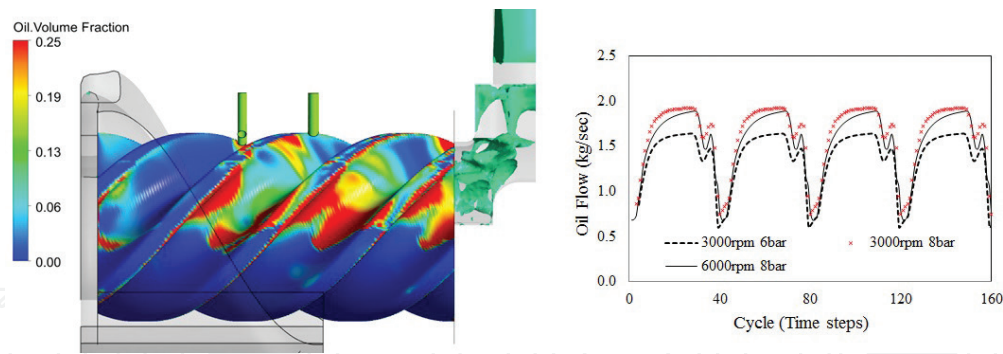


Figure 19. Oil injection interaction with the gate rotor lobes during one cycle.

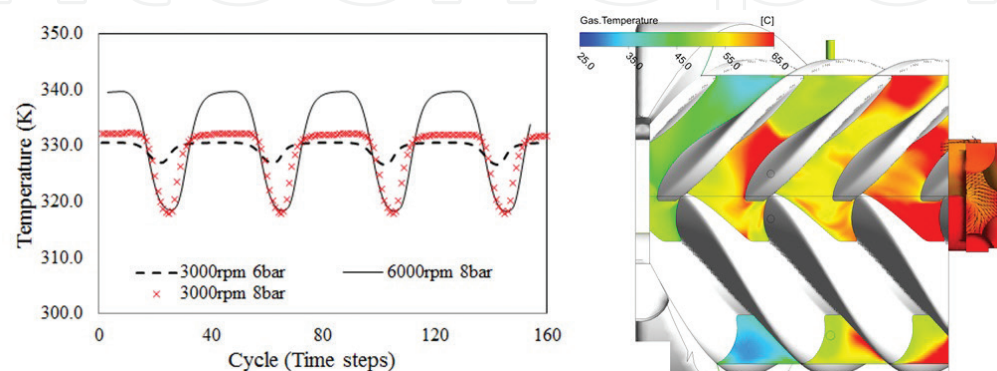


**Figure 20.** Oil flow distribution near the discharge port and flow cycles for three cases.

port can also be seen in **Figure 20**. Oil accumulates in the interlobe gaps and is transported by the rotors. The oil distribution in the discharge port is unsteady but cyclically repeating.

### 3.3. Gas temperature distribution

The cyclic variation of the discharge air temperature in the three cases is shown in **Figure 21**. With a built-in volume ratio of 3.6, the adiabatic discharge temperature without oil injection would be about 500 K, but due to leakage and recompression effects, the temperature rise would be much higher. Due to the oil being injected at 323 K, the gas temperature is substantially lowered and does not exceed 340 K. At 3000 rpm and 8 bar discharge pressure, the peak temperature is about 10° lower than that at 6000 rpm. This is because the amount of oil injected at the same discharge pressure and different rotational speeds is almost the same, as shown in **Figure 20**. This, when combined with the lower residence time of the oil at higher speed, there is less heat transfer from gas to oil, and this results in a higher discharge temperature. Similarly, there is only a small difference in the gas temperature at the same speed and different discharge pressures because the quantity of oil injected at higher pressure increases, as shown in **Figure 20**. This higher amount of oil lowers the gas temperature which remains close to the injection temperature of 323 K. The distribution of gas temperature at 8.0 bar discharge pressure and 6000 rpm is shown in **Figure 21**. It can be seen that the local maximum temperature reaches 340 K. Also, the gas temperature is not uniformly distributed. This non-uniformity is due to the non-homogeneous distribution of oil in the compression chamber.



**Figure 21.** Discharge gas temperature cycles for three cases and distribution at 6000 rpm (8.0 bar pressure).

## 4. Summary

In this chapter, the governing equations and strategies of mesh deformation have been reviewed for their applicability to twin-screw machine flow calculations. For the simplistic case of adiabatic compression and expansion by a piston in a cylinder type of system, the key-frame re-meshing method gave inconsistent results, with increasing error in pressure and temperature predictions in successive compression cycles, whereas the methods of diffusion smoothing and user-defined nodal displacement produced identical results with accurate estimates of pressure and temperature. Thus, it is necessary to extend the customised grid generation techniques and use a strategy such as user-defined nodal displacement, during the ALE calculations. A comparative study between the algebraic and differential approaches has helped in the selection of techniques that can offer feasibility for implementing grids for a deforming rotor domain and also for improving the grid quality. In the combined algebraic and differential approach, a special procedure has been introduced that completely smooths the transition of the algebraically generated splitting rack curve between the two rotors, thus smoothing the grid node movement in time and space. These grid features have improved the robustness of the solution in CFD solvers.

The results from the simulation of the compression process in an oil-injected twin-screw compressor provided an exceptional visualisation of the interaction of gas and oil inside the compression chamber. The interaction of the phases, the oil distribution and heat transfer between the gas and the oil and also the effects of high oil concentration in the leakage gaps on the sealing were well captured. The developments presented in this chapter together with the case study will be useful for flow analysts practising CFD in screw machines as well as computational solver developers considering such highly deforming grids in these classes of flow machines.

## Nomenclature

$x, y, z$	Cartesian coordinates
$q_{os}$	Flux source
$grad$	Gradient operator
$div$	Divergence operator
$\mathbf{v}$	Fluid velocity
$p$	Pressure
$V$	Cell volume
$\Delta t$	Time step
$\phi$	Conserved variable in transport equations
$\gamma$	Adiabatic index

SCORG	Screw Compressor Rotor Grid Generator
ALE	Arbitrary Lagrangian-Eulerian
$n$	Cell face normal vector
$q_{\phi V}$	Volume source
$S$	Cell surface
$\delta$	Displacement
$t$	Time
$T$	Temperature
$\mathbf{v}_b$	Surface velocity
$\Omega$	Control volume
$\Gamma$	Diffusion coefficient
$\rho$	Density
$\mu$	Viscosity
rms	Root-mean-square value

## Acknowledgements

A major portion of the chapter is derived from the thesis 'Grid Generation and CFD Analysis of Variable Geometry Screw Machines' [4].

## Author details

Sham Rane\*, Ahmed Kovačević, Nikola Stošić and Ian Smith

\*Address all correspondence to: [sham.rane@city.ac.uk](mailto:sham.rane@city.ac.uk)

School of Mathematics, Computer Science and Engineering, City, University of London, London, UK

## References

- [1] Rane S, Kovačević A. Algebraic generation of single domain computational grid for twin screw machines. Part I. Implementation. *Advances in Engineering Software*. 2017;**107**:38-50. DOI: <https://doi.org/10.1016/j.advengsoft.2017.02.003>



- [2] Kovačević A, Rane S. Algebraic generation of single domain computational grid for twin screw machines part II – Validation. *Advances in Engineering Software*. 2017;**109**:31-43. DOI: <https://doi.org/10.1016/j.advengsoft.2017.03.001>
- [3] Voorde VJ, Vierendeels J. A grid manipulation algorithm for ALE calculations in screw compressors. In: 17th AIAA Computational Fluid Dynamics Conference; Canada. Canada: AIAA; 2005. pp. 2005-4701
- [4] Rane S. Grid Generation and CFD Analysis of Variable Geometry Screw Machines [Thesis]. London: City, University of London; 2015. 350 p. Available from: <http://openaccess.city.ac.uk/id/eprint/15129>
- [5] Kovačević A. Three-Dimensional Numerical Analysis for Flow Prediction in Positive Displacement Screw Machines [Thesis]. London: City, University of London; 2002. 350 p
- [6] Kovačević A, Stošić N, Smith IK. *Screw Compressors – Three Dimensional Computational Fluid Dynamics and Solid Fluid Interaction*. 1st ed. New York: Springer-Verlag; 2007. 200 p. ISBN 3-540-36302-5
- [7] Rane S, Kovačević A, Stošić N. CFD analysis of oil flooded twin screw compressors. In: International Compressor Engineering Conference at Purdue; July; Purdue, USA. Purdue: 2016. p. 2392. DOI: <http://docs.lib.purdue.edu/icec/2392>
- [8] Rane S, Kovačević A, Stošić N, Kethidi M. Deforming grid generation and CFD analysis of variable geometry screw compressors. *Computers and Fluids*. 2014;**99**:124-141. DOI: <https://doi.org/10.1016/j.compfluid.2014.04.024>
- [9] Winslow A. Numerical solution of the quasilinear poisson equation in a nonuniform triangle mesh. *Journal of Computational Physics*. 1966;**1**(2):149-172. DOI: [https://doi.org/10.1016/0021-9991\(66\)90001-5](https://doi.org/10.1016/0021-9991(66)90001-5)
- [10] Ferziger JH, Perić M. *Computational Methods for Fluid Dynamics*. Berlin: Springer; 1996. ISBN 978-3-540-42074-3
- [11] Trulio GJ, Trigger RK. Numerical solution of the one-dimensional hydrodynamic equations in an arbitrary time-dependent coordinate system. University of California Lawrence Radiation Laboratory Report. 1961;UCLR-6522
- [12] Thomas P, Lombard C. Geometric conservation law and its application to flow computations on moving grids. *AIAA Journal*. 1979;**17**:1030-1037
- [13] Gosman DA. Prediction of in-cylinder processes in reciprocating internal combustion engines. *Computing Methods in Applied Science and Engineering*. 1984. Glowinski and Lions edition. pp. 604-629
- [14] Rienslagh K, Vierendeels J, Dick E. An arbitrary Lagrangian-Eulerian finite-volume method for the simulation of rotary displacement pump flow. *Applied Numerical Mathematics*. 2000;**32**:419-433